

JIANTING'S LECTURE REVIEW NOTES

最优化方法
Optimization Methods

2020 FALL

INSTRUCTOR : 李庆娜

TEXTBOOK : 最优化方法

REVIEW FOR OPTIMIZATION METHOD

Jianting Feng

29th November 2020

BEIJING INSTITUTE OF TECHNOLOGY

前言

这是笔者于 2020 年秋季学期于北京理工大学良乡校区学习最优化方法课程时的笔记总结，由于一些众所周知的原因，该笔记参考的主要书目并不为课程所指定的教材，而是：刘浩洋, 卢将, 李勇锋, 文再文, 最优化：建模、算法与理论, 高教出版社, 书号 978-7-04-055035-1 H. Liu, J. Hu, Y. Li, Z. Wen, Optimization: Model, Algorithm and Theory (in Chinese) 详情可参考文再文老师的主页。截止笔者完成该份笔记之前，本书还处于**即将出版**的状态。

另外一本个人觉得比较好的参考资料为 Algorithms for Optimization By Mykel J. Kochenderfer and Tim A. Wheeler。本书由 MIT 出版社出版，可以在 MIT Press 的官网购买该书并下载相应代码与 Slides（文老师的教材也可在相应网站下载代码）。

顺便吐槽一句，这课讲的东西真的不是最优化，证明都跟没讲一样，倒是幻灯片上有不少算例，这就算一个将数值优化方法的课吧。最后，LQNYYS!

Jianting Feng

2020 年 11 月

于北京理工大学

Contents

前言	i
1 预备知识 Preliminary	1
1.1 多元微积分	1
1.1.1 多元函数的 Taylor 展开	1
1.2 线性空间中的范数	1
1.3 凸集与凸函数	2
1.3.1 几个重要且常见的凸集	2
1.4 凸规划	3
1.5 收敛速度	4
2 线性规划 Linear Programming	5
3 无约束优化问题 Unconstrained Optimization	7
3.1 线搜索方法	7
3.1.1 精确线搜索算法	7
3.1.2 Armijo 准则	8
3.1.3 Wolfe-Powell 准则	9
3.1.4 线搜索法的收敛性分析	9
3.2 梯度类算法	11
3.2.1 梯度下降法	11
3.3 共轭方向法	13
3.4 共轭梯度法	15
3.4.1 Fletcher-Reeves 公式	15
3.5 牛顿法	16
3.6 拟牛顿法	17
3.6.1 割线方程	17
3.6.2 BFGS 公式	18
3.6.3 DFP 公式	18
4 约束优化问题 Constrained Optimization	19
4.1 约束最优化问题的最优性条件	19

5	补充内容一：锥及 Karush–Kuhn–Tucker 条件	23
5.1	凸锥 (Convex Cone)	23
5.1.1	法锥 (Normal Cone)	24
5.1.2	极锥 (Polar Cone)	24
5.1.3	对偶锥 (Dual Cone)	24
5.2	Lagrange 函数与 KKT 条件	25
5.2.1	Lagrange 函数	25
6	补充内容二：信赖域算法	29
6.1	算法简介	29
6.2	信赖域算法框架	29
6.3	信赖域算法子问题求解	32
6.3.1	Cauchy 点	32
6.3.2	dog leg 法	32

Chapter 1

预备知识 Preliminary

1.1 多元微积分

1.1.1 多元函数的 Taylor 展开

$$f(\mathbf{x}_0 + \mathbf{p}) = f(\mathbf{x}_0) + \nabla f(\mathbf{x}_0)^T \mathbf{p} + o(\|\mathbf{p}\|) \quad (1.1)$$

$$f(\mathbf{x}_0 + \mathbf{p}) = f(\mathbf{x}_0) + \nabla f(\mathbf{x}_0)^T \mathbf{p} + \frac{1}{2} \mathbf{p}^T \nabla^2 f(\mathbf{x}_0) \mathbf{p} + o(\|\mathbf{p}\|)^2 \quad (1.2)$$

若 $f(\mathbf{x}) = \mathbf{b}^T \mathbf{x}$, 则有

$$\nabla f(\mathbf{x}) = \mathbf{b}, \quad \nabla^2 f(\mathbf{x}) = \mathbf{0}_{n \times n} \quad (1.3)$$

若 $f(\mathbf{x}) = \frac{1}{2} \mathbf{x}^T Q \mathbf{x}$, 且 Q 为对称方阵, 则有

$$\nabla f(\mathbf{x}) = Q\mathbf{b}, \quad \nabla^2 f(\mathbf{x}) = Q \quad (1.4)$$

1.2 线性空间中的范数

Definition 1.2.1. 在 n 维线性空间 \mathbb{R}^n 中, 定义

$$\|\cdot\| : \mathbb{R}^n \rightarrow \mathbb{R}$$

满足下述三条公理:

1. (正定性) $\forall \mathbf{x} \in \mathbb{R}^n$, 有 $\|\mathbf{x}\| \geq 0$, 且 $\|\mathbf{x}\| = 0$ 当且仅当 $\mathbf{x} = \mathbf{0}$;
2. $\forall \mathbf{x} \in \mathbb{R}^n$ 及 $\alpha \in \mathbb{R}$, 有 $\|\alpha \mathbf{x}\| = |\alpha| \|\mathbf{x}\|$;
3. $\forall \mathbf{x}, \mathbf{y} \in \mathbb{R}^n$, 有 $\|\mathbf{x} + \mathbf{y}\| \leq \|\mathbf{x}\| + \|\mathbf{y}\|$

则称 $\|\cdot\|$ 为 \mathbb{R}^n 上的一个范数 (*norm*)。

Example 1.2.1. 常用的范数有如下几个：

$$1. \text{ 2-范数: } \|x\|_2 = \left(\sum_{i=1}^n x_i^2\right)^{\frac{1}{2}}$$

$$2. \text{ } \infty\text{-范数: } \|x\|_\infty = \max_{1 \leq i \leq n} |x_i|$$

$$3. \text{ 1-范数: } \|x\|_1 = \sum_{i=1}^n |x_i|$$

1.3 凸集与凸函数

Definition 1.3.1. 设 $D \subseteq \mathbb{R}^n$ ，若对于任意的 $\lambda \in [0, 1]$ ，都有

$$\lambda x + (1 - \lambda)y \in D,$$

则称 D 为**凸集**。称 $\lambda x + (1 - \lambda)y$ 为凸组合。

Note 1.3.1. 用数学归纳法，容易证明对于任意有限个元素的凸组合仍满足上述性质。

1.3.1 几个重要且常见的凸集

超平面与半空间

对于任意非零向量 a ，我们称形如 $\{x \mid a^T x = b\}$ 的集合为**超平面**，形如 $\{x \mid a^T x \leq b\}$ 的集合为**半空间**。

球、椭球、锥

Definition 1.3.2. 设 $\|\cdot\|$ 为一个范数，定义

$$B(x_c, r) = \{x \mid \|x - x_c\| \leq r\}$$

为半径为 r ，中心为 x_c 的**球**。

Definition 1.3.3. 形如

$$\{x \mid (x - x_c)^T P^{-1} (x - x_c) \leq 1\}$$

中心为 x_c 的**椭球**，其中 $P \in \mathcal{S}_{++}^n$ （即 P 对称正定）。

Definition 1.3.4. 称集合

$$\{(x, t) \mid \|x\| \leq t\}$$

为**范数锥**，*Euclid* 范数锥也称**二次锥**。

更多与锥（cone）相关的内容，请参考5

Definition 1.3.5 (凸函数). 设 $f(x)$ 为适当函数，如果 f 的定义域 D 为凸集，且满足

$$f(\lambda x + (1 - \lambda)y) \leq \lambda f(x) + (1 - \lambda)f(y)$$

对于所有 $x, y \in D$ 与 $0 \leq \lambda \leq 1$ 都成立，则称 f 是**凸函数**。

Definition 1.3.6 (严格凸函数). 设 $f(x)$ 为适当函数, 如果 f 的定义域 D 为凸集, 且满足

$$f(\lambda x + (1 - \lambda)y) < \lambda f(x) + (1 - \lambda)f(y)$$

对于所有 $x, y \in D$ 与 $0 < \lambda < 1$ 都成立, 则称 f 是**严格凸函数**。

Theorem 1.3.1 (凸函数的判定定理). $f(x)$ 是定义在凸集 $D \subset \mathbb{R}^n$ 上的 (严格) 凸函数, 则对于任意 $x, y \in D$, 令

$$\phi(t) = f(tx + (1 - t)y), \quad t \in [0, 1]$$

$\phi(t)$ 为 $[0, 1]$ 上的 (严格) 凸函数。

Note 1.3.2. 事实上, 上面定理的说的是, 凸函数的一个最基本的判定方式是: 先将其限制在任意直线上, 然后判断对应的一维函数是否是凸的。

对于可微函数, 我们判断其是否为凸函数还可以根据它的导数信息来判断

Theorem 1.3.2 (一阶条件). 对于定义在凸集 $D \subset \mathbb{R}^n$ 上的可微函数 f , f 是凸函数当且仅当下式成立

$$f(y) \geq f(x) + \nabla f(x)^T(y - x) \quad \forall x, y \in D.$$

Theorem 1.3.3 (二阶条件). 对于定义在凸集 $D \subset \mathbb{R}^n$ 上的二阶连续可微函数 f , f 是凸函数当且仅当下式成立

$$\nabla^2 f(x) \succeq 0 \quad \forall x \in D.$$

如果 $\nabla^2 f(x) \succ 0 \quad \forall x \in D$, 则 f 是严格凸函数。

Definition 1.3.7 (水平集). 设函数 $f(x)$ 定义在 D 上,

$$D_\alpha := \{x \mid x \in D, f(x) \leq \alpha\}$$

称为 $f(x)$ 的**水平集**。

由水平集的定义及凸函数的判定性质, 立即有

Theorem 1.3.4. 凸集上的凸函数的水平集为凸集, 即设凸函数 $f(x)$ 定义在凸集 $D \subset \mathbb{R}^n$ 上, 则

$$D_\alpha = \{x \mid x \in D, f(x) \leq \alpha\}$$

为凸集。

1.4 凸规划

设凸函数 $f(x)$ 定义在凸集 $D \subset \mathbb{R}^n$ 上, 则下列规划问题

$$\begin{aligned} & \min f(x) \\ & s.t. \begin{cases} g_i(x) \geq 0, & i = 1, 2, \dots, m \\ h_j(x) = 0, & j = 1, 2, \dots, l \end{cases} \end{aligned}$$

若 $g_i(x)$ 为凸函数, $h_j(x)$ 为线性函数, 则上述问题称为**凸规划**。

1.5 收敛速度

Definition 1.5.1. 设序列 $\{\mathbf{x}_k\}$ 收敛于 \mathbf{x}^* , 且有

$$\lim_{k \rightarrow \infty} \frac{\|\mathbf{x}_{k+1} - \mathbf{x}^*\|}{\|\mathbf{x}_k - \mathbf{x}^*\|} = \beta$$

1. 若 $\beta = 1$, 则称该序列**次线性收敛**
2. 若 $0 < \beta < 1$, 则称该序列**线性收敛**, β 被称为**收敛比**
3. 若 $\beta = 0$, 则称该序列**超线性收敛**

Definition 1.5.2. 设序列 $\{\mathbf{x}_k\}$ 收敛于 \mathbf{x}^* , 若对于某个实数 $p \geq 1$, 有

$$\lim_{k \rightarrow \infty} \frac{\|\mathbf{x}_{k+1} - \mathbf{x}^*\|}{\|\mathbf{x}_k - \mathbf{x}^*\|^p} = \beta, \quad 0 < \beta < +\infty$$

则称该序列为 p 阶收敛的.

Chapter 2

线性规划 Linear Programming

Chapter 3

无约束优化问题 Unconstrained Optimization

本节考虑如下的无约束优化问题：

$$\min_{x \in \mathbb{R}^n} f(x), \quad (3.1)$$

其中 $f(x) : \mathbb{R}^n \rightarrow \mathbb{R}$

3.1 线搜索方法

线搜索算法的数学表述为：给定当前点 x^k ，首先通过某种算法（稍后会涉及）选取**下降方向** d^k ，确定**步长** $\alpha_k > 0$ ，则下一步的迭代点可以写作

$$x^{k+1} = x^k + \alpha_k d^k \quad (3.2)$$

d^k 是下降方向的要求为 $(d^k)^T \nabla f(x^k) < 0$ ，即要求了沿该方向函数值 f 会减小。线搜索回答了如何选择一个合适的步长 α_k 。

实际上这是一个一维的优化问题，我们构造辅助函数

$$\phi(\alpha) = f(x^k + \alpha d^k)$$

显然，这是将 $f(x)$ 限制在 $\{x^k + \alpha d^k : \alpha > 0\}$ 一个一维区域上。线搜索的目的实际上是选取合适的 α_k 使得 $\phi(\alpha_k)$ 足够小，于是我们有两种想法：找到使函数值在该方向上的最小值或找到一个能够使函数值下降的步长（不需要下降到最小值），分别称为**精确线搜索算法**与**非精确线搜索算法**。

3.1.1 精确线搜索算法

正如上面所讨论的相同，我们是求解如下的问题

$$\alpha_k = \arg \min_{\alpha > 0} \phi(\alpha),$$

即 α_k 为最佳步长，这实际上可能需要极大的计算量（针对光滑函数可以想象为求导函数的根，详情参见数值计算方法中求解非线性方程），所以我们在实际应用中很少使用这种方法，我们仅需要 $\phi(\alpha_k)$ 满足一些不等式性质，这样产生的算法就是我们下面要介绍的非精确搜索算法

3.1.2 Armijo 准则

首先，我们引入一个较为常用的线搜索准则，它保证了每一步迭代充分下降.

Definition 3.1.1 (Armijo 准则). 设 d^k 为点 x^k 处的下降方向，若

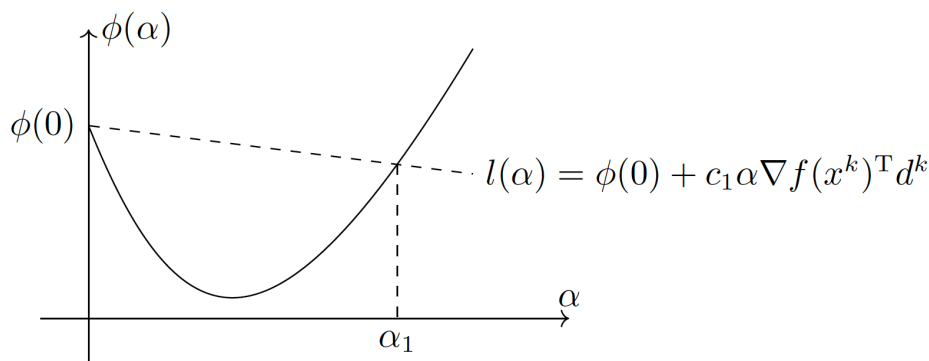
$$f(x^k + \alpha d^k) \leq f(x^k) + c_1 \alpha \nabla f(x^k)^T d^k,$$

则称步长 α 满足 **Armijo 准则**，其中 $c_1 \in (0, 1)$ 是一个预先给定的常数.

Armijo 准则的几何意义也是很明显的，参见下图 它的含义是指点 $(\alpha, \phi(\alpha))$ 必须在直线

Figure 3.1

Armijo 准则



$$l(\alpha) = \phi(0) + c_1 \alpha \nabla f(x^k)^T d^k$$

的下方，由于 d^k 选取的是下降方向，于是 $\nabla f(x^k)^T d^k < 0$ ，该直线的斜率一定为负. 通常应用中取 c_1 足够小，这样使得 Armijo 条件非常容易满足，但是，如果选取的步长太小，那么迭代的点可能固定不变.

在实际使用过程中，我们常常将 Armijo 算法与**回退法**一同使用. 给定初值 $\hat{\alpha}$ ，回退法通过不断以指数方式缩小来试探步长，找到的第一个满足 Armijo 准则的点.

Algorithm 1 线搜索回退法

- 1: 选择初始步长 $\hat{\alpha}$ ，参数 $\gamma, c \in (0, 1)$. 初始化 $\alpha \leftarrow \hat{\alpha}$.
 - 2: **while** $f(x^k + \alpha d^k) > f(x^k) + c_1 \alpha \nabla f(x^k)^T d^k$, **do**
 - 3: 令 $\alpha \leftarrow \gamma \alpha$
 - 4: **end while**
 - 5: 输出 $\hat{\alpha} = \alpha$
-

3.1.3 Wolfe-Powell 准则

Wolfe-Powell 准则也叫做 Wolfe 准则或 Armijo-Wolfe 准则.

Definition 3.1.2 (Wolfe 准则). 设 d^k 是点 x^k 处的下降方向, 若

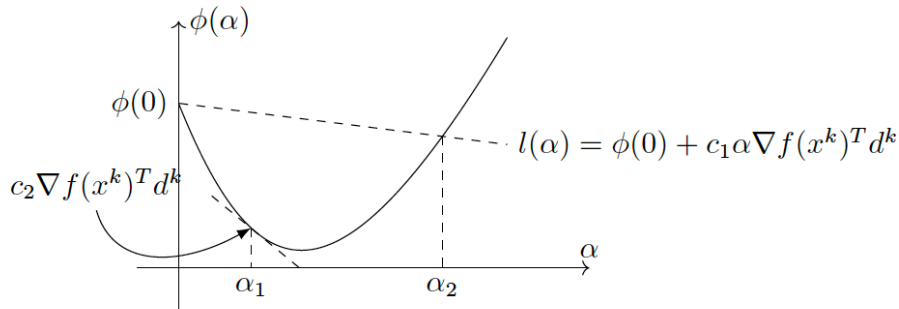
$$\begin{aligned} f(x^k + \alpha d^k) &\leq f(x^k) + c_1 \alpha \nabla f(x^k)^T d^k, \\ \nabla f(x^k + \alpha d^k)^T d^k &\geq c_2 \nabla f(x^k)^T d^k, \end{aligned} \quad (3.3)$$

则称步长 α 满足 **Wolfe 准则**, 其中 $c_1, c_2 \in (0, 1)$ 为给定常数且有 $c_1 < c_2$.

准则中第一个不等式即为 Armijo 准则, 而第二个不等式为 Wolfe 准则的要求. 注意到 $\nabla f(x^k + \alpha d^k)^T d^k$ 恰好为 $\phi(\alpha)$ 的导数, Wolfe 准则实际要求 $\phi(\alpha)$ 在 α 处的切线斜率不能小于 $\phi'(0)$ 的 c_2 倍. 如图所示, 在 $[\alpha_1, \alpha_2]$ 中的点均满足 Wolfe 准则, 并且在极小值点 α^* 处有 $\phi(\alpha^*) = \nabla f(x^k + \alpha^* d^k)^T d^k = 0$, 于是 α^* 永远满足第二个不等式, 只需选择较小的 c_1 满足第一个不等式即可.

Figure 3.2

Wolfe 准则



3.1.4 线搜索法的收敛性分析

Theorem 3.1.1 (Zoutendijk). 考虑一般的迭代格式(3.2). 其中 d^k 是搜索发现, α_k 是步长, 且在迭代的过程中满足 Wolfe 准则. 假设目标函数 f 下有界、连续可微且梯度 ∇f Lipschitz 连续, 即

$$\|\nabla f(x) - \nabla f(y)\| \leq L\|x - y\|, \quad \forall x, y \in \mathbb{R}^n,$$

那么我们有

$$\sum_{k=0}^{\infty} \cos^2 \theta_k \|\nabla f(x^k)\|^2 \leq +\infty$$

其中 $\cos \theta_k$ 为负梯度 $-\nabla f(x^k)$ 和下降方向 d^k 夹角的余弦, 即

$$\cos \theta_k = \frac{-\nabla f(x^k)^T d^k}{\|\nabla f(x^k)\| \|d^k\|}$$

这个定理的证明是某老师喜欢考的，所以我们在此仔细加以讨论

Proof. 由 Wolfe 准则的第二个不等式，我们有

$$(\nabla f(x^{k+1}) - \nabla f(x^k))^T d^k \geq (c_2 - 1) \nabla f(x^k)^T d^k.$$

有 Cauchy-Schwarz 不等式与 Lipschitz 条件

$$\begin{aligned} (\nabla f(x^{k+1}) - \nabla f(x^k))^T d^k &\leq \|\nabla f(x^{k+1}) - \nabla f(x^k)\| \|d^k\| \\ &\leq L \|x^{k+1} - x^k\| \|d^k\| \\ &\leq \alpha_k L \|d^k\|^2. \end{aligned}$$

结合上面俩式，有

$$(c_2 - 1) \nabla f(x^k)^T d^k \leq \alpha_k L \|d^k\|^2$$

即步长 α_k 满足

$$\alpha_k \geq \frac{(c_2 - 1) \nabla f(x^k)^T d^k}{L \|d^k\|^2}$$

将此式代入 Wolfe 准则的第一个不等式，得到

$$\begin{aligned} f(x^{k+1}) &\leq f(x^k) + c_1 \frac{(c_2 - 1) \nabla f(x^k)^T d^k}{L \|d^k\|^2} \nabla f(x^k)^T d^k \\ &= f(x^k) + \frac{c_1(c_2 - 1)}{L} \frac{(\nabla f(x^k)^T d^k)^2}{\|d^k\|^2} \\ &= f(x^k) - \frac{c_1(1 - c_2)}{L} \cos^2 \theta_k \|\nabla f(x^k)\|^2. \end{aligned}$$

这显然是一个递推的不等式，右侧关于 k 递推（从 k 到 0），得到

$$f(x^{k+1}) \leq f(x^0) - \frac{c_1(1 - c_2)}{L} \sum_{j=0}^k \cos^2 \theta_j \|\nabla f(x^j)\|^2$$

由于 f 是下有界且下降的（Wolfe 准则保证了严格下降性），同时由于 $0 < c_1 < c_2 < 1$ ，有 $c_1(1 - c_2) > 0$ ，于是我们得到了

$$\sum_{j=0}^k \cos^2 \theta_j \|\nabla f(x^j)\|^2 < +\infty$$

令 $k \rightarrow \infty$ ，得到我们所需要的

$$\sum_{j=0}^{\infty} \cos^2 \theta_j \|\nabla f(x^j)\|^2 < +\infty \quad (3.4)$$

□

该定理指出了在满足 Wolfe 准则，对梯度 Lipschitz 连续且下有界的函数总能得到(3.4)条件成立。Zoutendijk 定理实际上刻画了线搜索准则的性质，配合一定的下降方向 d^k 选取方式，我们能够得到线搜索最基本的收敛性。

Theorem 3.1.2 (线搜索算法的收敛性). 对于迭代格式(3.2), 设 θ_k 为每一步迭代过程中下降方向 d^k 与负梯度方向 $-\nabla f(x^k)$ 的夹角, 并且假设对于任意的 k , 都存在常数 γ 使得

$$\theta_k < \frac{\pi}{2} - \gamma$$

严格成立, 则在(3.4)条件成立的前提下, 我们能得到算法一定是收敛的, 即

$$\lim_{k \rightarrow \infty} \nabla f(x^k) = 0.$$

Proof. 直接证明较为困难, 我们考虑反证法, 假设存在子列 $\{k_l\}$ 和正常数 $\delta > 0$, 使得结论不成立, 即

$$\|\nabla f(x^{k_l})\| \geq \delta, \quad l = 1, 2, \dots,$$

根据 θ_k 的假设, 对任意 k , 有

$$\cos \theta_k > \sin \gamma > 0.$$

于是

$$\begin{aligned} \sum_{k=0}^{\infty} \cos^2 \theta_k \|\nabla f(x^k)\|^2 &\geq \sum_{l=1}^{\infty} \cos^2 \theta_{k_l} \|\nabla f(x^{k_l})\|^2 \\ &\geq \sum_{l=1}^{\infty} (\sin^2 \gamma) \cdot \delta^2 \rightarrow +\infty \end{aligned}$$

这与(3.4)矛盾, 因此一定有

$$\lim_{k \rightarrow \infty} \nabla f(x^k) = 0.$$

□

Note 3.1.1. 这个推论的几何意义是显然的, 如果下降方向 d^k 与梯度方向正交的话, 根据 *Taylor* 公式的一阶近似, 函数值 $f(x^k)$ 几乎不发生改变, 因此我们在后面求解下降方向时, 总是要求它与梯度的垂直方向的夹角有一定的下界, 这保证了我们在求解步长时的收敛性.

3.2 梯度类算法

梯度类算法实际上是采用了函数的一阶信息 (即梯度 Gradient), 我们在这里只介绍梯度下降法与共轭梯度法.

3.2.1 梯度下降法

对于光滑函数 $f(x)$, 迭代至 x^k 处, 我们要选取一个合适的方向 d^k 作为下降方向. 采用之前的记号 $\phi(\alpha) = f(x^k + \alpha d^k)$, 在 $\alpha = 0$ 附近的一阶带 Peano 余项的 Taylor 展开

$$\phi(\alpha) = \phi(0) + \alpha \nabla f(x^k)^T d^k + \mathcal{O}(\alpha^2 \|d^k\|^2),$$

由 Cauchy-Schwarz 不等式, 当 α 足够小时, 选取 $d^k = -\nabla f(x^k)$ 可使函数值下降最快, 于是我们有了梯度下降法的迭代格式

$$x^{k+1} = x^k - \alpha_k \nabla f(x^k). \quad (3.5)$$

其中的 α_k 既可以取为常数也可以采取一定算法取自适应步长.

在介绍具体的算法之前, 我们先分析该算法对对称正定的二次函数的收敛速度

Theorem 3.2.1. 考虑二次正定函数

$$f(x) = \frac{1}{2}x^T A x - b^T x,$$

其最优点 x^* , 若使用梯度下降法(3.5)并选取 α_k 为精确线搜索的步长, 即

$$\alpha_k = \frac{\|\nabla f(x^k)\|^2}{\nabla f(x^k)^T A \nabla f(x^k)},$$

那么梯度下降法是关于迭代点列 $\{x^k\}$ 是 Q -线性收敛 (平方), 即

$$\|x^{k+1} - x^*\|_A^2 \leq \left(\frac{\lambda_1 - \lambda_n}{\lambda_1 + \lambda_n} \right)^2 \|x^k - x^*\|_A^2$$

其中 λ_1, λ_n 分别为 A 的最大、最小特征值, $\|x\|_A \triangleq \sqrt{x^T A x}$ 为正定矩阵 A 诱导的范数.

Note 3.2.1. 我们可以推出如下的式子

$$\frac{\|x^{k+1} - x^*\|_A}{\|x^k - x^*\|_A} \leq \frac{|\lambda_1 - \lambda_n|}{|\lambda_1 + \lambda_n|}$$

取极限有, 即在 $\|\cdot\|_A$ 范数意义下, 迭代格式(3.5)至少是线性收敛的. 同时收敛比有一个上界

$$\beta \leq \frac{|\lambda_1 - \lambda_n|}{|\lambda_1 + \lambda_n|}$$

Note 3.2.2. 我们同时能发现, 该迭代方法的收敛速度与矩阵的谱 (Spectrum) 的分布有关, 即特征值分布越密集, 收敛越快. 这个描述的几何意义是明显的, 当所有特征值都接近时, $\frac{|\lambda_1 - \lambda_n|}{|\lambda_1 + \lambda_n|}$ 的值达到最小值, 即该二次函数的等高线近似为中心相同的一组圆.

下面我们给出梯度下降法具体的算法,

Algorithm 2 梯度下降法

Require: 控制误差 $\varepsilon > 0$

- 1: 选择初始点 x^0 . 初始化 $k = 0$.
- 2: **while** $\|\nabla f(x^k)\| \geq \varepsilon$ **do**
- 3: 令下降方向 $d^k \leftarrow -\nabla f(x^k)$
- 4: 做一维搜索

$$\alpha_k = \arg \min_{\alpha \geq 0} f(x^k + \alpha d^k)$$

- 5: 令 $x^{k+1} \leftarrow x^k + \alpha_k d^k$, $k \leftarrow k + 1$
 - 6: **end while**
 - 7: 输出 $x^* = x^k$
-

Note 3.2.3. 对于做精确一维搜索的对称正定二次函数，迭代格式如下

$$x^{k+1} = x^k - \frac{\|\nabla f(x^k)\|^2}{\nabla f(x^k)^T A \nabla f(x^k)} \nabla f(x^k)$$

关于做精确一维搜索的梯度下降法，两次搜索的方向是彼此正交的，即有如下的命题

Proposition 3.2.1. 设 $\{d^k\}$ 是由做精确一维搜索的梯度下降法产生的方向向量序列，则有

$$\langle d^k, d^{k+1} \rangle = 0$$

其中 $\langle \cdot, \cdot \rangle$ 为向量内积.

Proof. 引入之前的记号，

$$\phi(\alpha) = f(x^k + \alpha d^k)$$

由于做精确的一维搜索，我们不妨假设 f 足够光滑，同时注意到 α 为最优步长，则有

$$\phi'(\alpha) = \nabla f(x^k + \alpha d^k)^T d^k = 0$$

而 $d^{k+1} = \nabla f(x^k + \alpha d^k)$ 为在 $k+1$ 次迭代时的方向，于是我们得到了

$$\langle d^k, d^{k+1} \rangle = 0$$

正是我们需要的. □

Note 3.2.4. 由于梯度下降法的相邻两次搜索方向闭相交，于是在接近最优解的地方可能出现 Zig-zag 现象，即走“之”字型路线，收敛性不够好.

3.3 共轭方向法

在介绍共轭梯度法之前，我们先引入二次终止性的定义

Definition 3.3.1 (二次终止性 (Quadratic Termination)). 如果一个算法对二次正定函数经过有限次迭代可以得到极小值点，这种算法具有二次终止性 (Quadratic Termination).

Definition 3.3.2 (共轭向量). 设 G 为 n 阶正定矩阵， p^1, p^2, \dots, p^k 为 n 维向量组，若

$$(p^i)^T G p^j = 0, \quad i, j = 1, 2, \dots, k, i \neq j$$

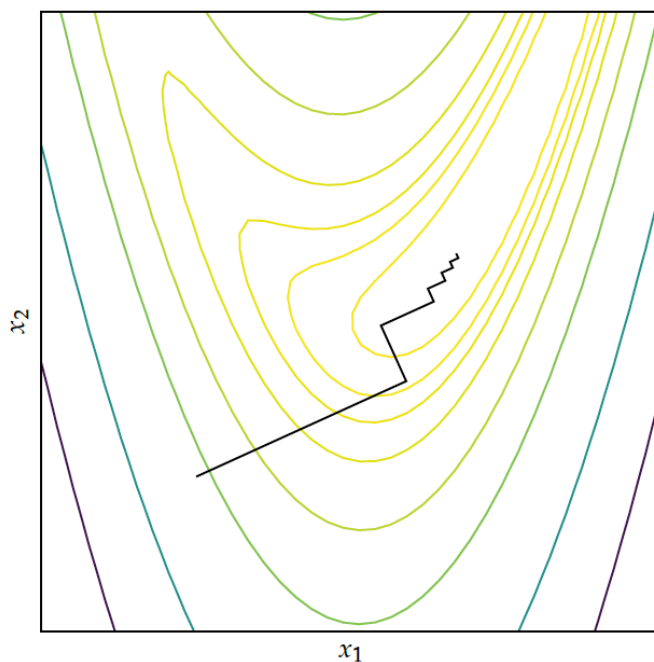
则称向量组 p^1, p^2, \dots, p^k 关于 G 共轭

Note 3.3.1. 实际上，由矩阵特征向量 (Eigenvector) 的性质，我们容易有如下结果：

1. 共轭向量组线性无关，
2. 共轭向量 p^1, p^2, \dots, p^n 构成 \mathbb{R}^n 的一组基 (思考为什么这里 $k = n$)，

Figure 3.3

Zig zag 现象



3. 求解共轭向量组实际上是求解矩阵特征向量.

待求解的问题为二次问题

$$f(x) = \frac{1}{2}x^T Gx + b^T x + c \quad (3.6)$$

进行 n 步搜索, 由于 n 个共轭向量彼此线性无关, 构成 \mathbb{R}^n 的一组基, 故理论上经过 n 步就可以对二次正定函数得到最小值点.

Algorithm 3 共轭方向法

Require: 控制误差 $\varepsilon > 0$

1: 选择初始点 x^0 及初始下降方向 p^0 . 初始化 $k \leftarrow 0$.

2: **while** $\|g^k\| \geq \varepsilon$ **do**

3: 做一维精确线搜索, 确定步长

$$\alpha_k = \arg \min_{\alpha \geq 0} f(x^k + \alpha p^k)$$

4: 令 $x^{k+1} \leftarrow x^k + \alpha_k d^k$, $k \leftarrow k + 1$

5: 取共轭方向 p^{k+1} , 使得

$$p^{k+1} G p^i = 0, \quad i = 1, 2, \dots, k.$$

6: **end while**

7: 输出 $x^* = x^k$

下面给出求共轭方向的算法，类似 Gram-Schmidt 正交化方法，记

$$\langle a, b \rangle_G = a^T G b$$

为正定矩阵 G 诱导出的内积，容易验证其满足内积的定义（实际上就是将度量矩阵由 I 换为了对称正定矩阵 G ）。

$$\begin{aligned} q^1 &= p^1 \\ q^2 &= p^2 - \frac{\langle p^2, q^1 \rangle_G}{\langle p^1, q^1 \rangle_G} q^1 \\ &\dots \\ q^n &= p^n - \sum_{i=1}^{n-1} \frac{\langle p^n, q^i \rangle_G}{\langle p^i, q^i \rangle_G} q^i. \end{aligned}$$

下面介绍一种更简易的构造共轭方向的方法，由于需要用到梯度，所以被称为共轭梯度法。

3.4 共轭梯度法

Algorithm 4 共轭梯度法

Require: 控制误差 $\varepsilon > 0$

1: 选择初始点 x^0 及初始下降方向 p^0 . 初始化 $k \leftarrow 0$.

2: $p_0 \leftarrow -g_0$

3:

$$\beta_0 \leftarrow \frac{(g^1)^T G p^0}{(p^0)^T G p^0}$$

4: **while** $\|g^k\| \geq \varepsilon$ **do**

5: 令 $p^{k+1} = -g^{k+1} + \beta_k p^k$, $k \leftarrow k + 1$

6: $x^{k+1} \leftarrow x^k$

7:

$$\beta_k \leftarrow \frac{(p^{k+1})^T G p^k}{(p^k)^T G p^k}$$

8: **end while**

9: 输出 $x^* = x^k$

由于一般的函数求 G (Hessian 阵) 计算量较大，于是我们给出几种不显含 G 的表达式。

$$\beta_{k-1} = \frac{(g^k)^T G p^{k-1}}{(p^{k-1})^T G p^{k-1}}$$

3.4.1 Fletcher-Reeves 公式

由于

$$G p^{k-1} = G \frac{1}{\alpha_{k-1}} (x^k - x^{k-1}) = \frac{1}{\alpha_{k-1}} (g^k - g^{k-1})$$

于是

$$(g^k)^T G p^{k-1} = \frac{1}{\alpha_{k-1}} (g^k)^T (g^k - g^{k-1}) = \alpha_{k-1} (g^k)^T g^k$$

同时

$$(p^{k-1})^T G p^{k-1} = \frac{1}{\alpha_{k-1}} (-g^{k-1} + \beta_{k-2} p^{k-2})^T (g^k - g^{k-1}) = \frac{1}{\alpha_{k-1}} (g^{k-1})^T g^{k-1}$$

故

$$\beta_{k-1} = \frac{(g^k)^T g^k}{(g^{k-1})^T g^{k-1}}$$

Algorithm 5 FR 共轭梯度法

Require: 控制误差 $\varepsilon > 0$

1: 选择初始点 x^0 . 初始化 $k \leftarrow 0$.

2: **while** $\|g^k\| \geq \varepsilon$ **do**

3:

$$\beta_{k-1} = \begin{cases} \frac{(g^k)^T g^k}{(g^{k-1})^T g^{k-1}}, & k \geq 1 \\ 0, & k = 0. \end{cases}$$

$$p^k = -g^k + \beta_{k-1} p^{k-1}$$

4: 由一维搜索确定步长 α_k

5: 令 $x^{k+1} \leftarrow x^k + \alpha_k p^k$, $k \leftarrow k + 1$

6: **end while**

7: 输出 $x^* = x^k$

3.5 牛顿法

本节我们讨论经典牛顿法, 牛顿法属于二阶算法. 对于二次连续可微函数 $f(x)$, 考虑其在 x^k 处的二阶 Taylor 展开

$$f(x^k + d^k) = f(x^k) + \nabla f(x^k)^T d^k + \frac{1}{2} (d^k)^T \nabla^2 f(x^k) d^k + \mathcal{O}(\|d^k\|^2)$$

将函数看做对 d^k 的极小化问题, 并忽略高阶项, 得到

$$h(d^k) = f(x^k) + \nabla f(x^k)^T d^k + \frac{1}{2} (d^k)^T \nabla^2 f(x^k) d^k$$

求导

$$\frac{\partial h}{\partial d^k} = \nabla f(x^k) + \nabla^2 f(x^k) d^k = 0$$

得到 d^k 所满足的条件为

$$\nabla^2 f(x^k) d^k = -\nabla f(x^k) \tag{3.7}$$

上式(3.7)也被称为牛顿方程, 容易得到当 $\nabla^2 f(x^k)$ 非奇异时, 可以取 $d^k = -\nabla^2 f(x^k)^{-1} \nabla f(x^k)$. 一般满足式(3.7)的方向也被称为**牛顿方向**. 经典牛顿法步长恒为 1, 因此其迭代格式为

$$x^{k+1} = x^k - \nabla^2 f(x^k)^{-1} \nabla f(x^k).$$

如果步长可变, 则称之为阻尼牛顿法.

3.6 拟牛顿法

牛顿法在理论和实践上均有很好的效果, 但由于其用到了二阶的 Hessian 阵, 导致其对大规模问题的效果不是很好, 于是我们猜想能否构造 Hessian 矩阵的近似矩阵, 但任然保持一定的性质. 我们接下来就将推导这些性质.

3.6.1 割线方程

回顾对牛顿法的推导过程, 我们有

$$\nabla f(x) = \nabla f(x^{k+1}) + \nabla^2 f(x^{k+1})(x - x^{k+1}) + \mathcal{O}(\|x - x^{k+1}\|^2).$$

令 $x = x^k, s^k = x^{k+1} - x^k, y^k = \nabla f(x^{k+1}) - \nabla f(x^k)$, 并忽略高阶项, 得到

$$\nabla^2 f(x^{k+1}) s^k = y^k$$

我们希望 Hessian 矩阵的近似矩阵 B^{k+1} 也满足同样的方程, 即

$$y^k = B^{k+1} s^k, \tag{3.8}$$

或者其逆矩阵 H^{k+1} 近似满足

$$s^k = H^{k+1} y^k, \tag{3.9}$$

并称式(3.8)与(3.9)为**割线方程**.

Algorithm 6 拟牛顿法算法框架

Require: 初始点 $x^0 \in \mathbb{R}^n$, 初始矩阵 $B^0 \in \mathbb{R}^{n \times n}$ (或 H^0).

- 1: 初始化 $k \leftarrow 0$.
 - 2: **while** 未达到停机准则 **do**
 - 3: 计算方向 $d^k = -(B^k)^{-1} \nabla f(x^k)$ 或 $d^k = -H^k \nabla f(x^k)$.
 - 4: 通过线搜索找到合适步长 $\alpha_k > 0$, 令 $x^{k+1} \leftarrow x^k + \alpha_k d^k$
 - 5: 更新 Hessian 矩阵的近似矩阵 B^{k+1} 或其逆 H^{k+1} .
 - 6: $k \leftarrow k + 1$
 - 7: **end while**
-

实际使用过程中找 H^k 更为常用, 因为这种方法不要求解线性方程组.

实际上课本中介绍的 BFGS 公式 (秩二更新) 是为了改进秩一更新的部分缺点而提出的, 详情还是请大家参考文再文老师的教材, 这里就不展开细说了.

3.6.2 BFGS 公式

我们考虑对 Hessian 阵近似矩阵 B^k 的秩二更新, 采用待定系数法来推导公式, 设

$$B^{k+1} = B^k + auu^T + bvv^T, \quad (3.10)$$

其中 $u, v \in \mathbb{R}^n$, $a, b \in \mathbb{R}$ 待定. 根据割线方程(3.8),

$$B^{k+1}s^k = (B^k + auu^T + bvv^T)s^k y^k,$$

整理得到

$$(a \cdot u^T s^k)u + (b \cdot v^T s^k)v = y^k - B^k s^k.$$

通过选取 u 和 v 让上述等式成立即可, 一个较为显然的选法为对应项分别相等, 即

$$\begin{aligned} u &= y^k, & a \cdot u^T s^k &= 1, \\ v &= B^k s^k, & b \cdot v^T s^k &= -1. \end{aligned}$$

因此得到更新方式

$$B^{k+1} = B^k + \frac{y^k(y^k)^T}{(s^k)^T y^k} - \frac{B^k s^k (B^k s^k)^T}{(s^k)^T B^k s^k}. \quad (3.11)$$

即为 BFGS 公式, 它是由 Broyden, Fletcher, Goldfarb, Shanno 四人名字的首字母组成

3.6.3 DFP 公式

在 BFGS 公式的推导过程中, 如果利用割线方程(3.9)来对 H^k 进行秩二修正, 得到基于 H^k 的拟牛顿矩阵更新

$$H^{k+1} = H^k - \frac{H^k y^k (H^k y^k)^T}{(y^k)^T H^k y^k} + \frac{s^k (s^k)^T}{(y^k)^T s^k}. \quad (3.12)$$

上式即为 DFP 公式, 这种迭代格式首先由 Davidon 发现, 此后由 Fletcher 以及 Powell 进一步发展, 因此被称为 DFP 公式.

Chapter 4

约束优化问题 Constrained Optimization

本章内容主要研究的是约束最小化问题：

$$\begin{aligned} \min_{x \in \mathbb{R}} f(x) \\ s.t. \begin{cases} c_i(x) = 0, i \in \mathcal{E}, \\ c_i(x) \geq 0, i \in \mathcal{I}, \end{cases} \end{aligned} \quad (4.1)$$

约定 c_i 为定义在 \mathbb{R} 或其子集上的实值函数， \mathcal{E} 和 \mathcal{I} 分别是等式约束和不等式约束的指标集。于是该问题的可行域为

$$\mathcal{X} = \{x \in \mathbb{R} \mid c_i(x) = 0, i \in \mathcal{E} \text{ 且 } c_i(x) \geq 0, i \in \mathcal{I}\}.$$

一个较为明显的做法是将 \mathcal{X} 的示性函数加到目标函数后组成无约束优化问题，但这样造成函数的性质并不是很好。为解决该问题，我们首先要给出约束优化问题的最优性理论，为此我们引入如下定义。

4.1 约束最优化问题的最优性条件

Definition 4.1.1 (切锥 (tangent cone)). 给定可行域 \mathcal{X} 及其内部一点 x ，若存在可行序列 $\{z_k\}_{k=1}^{\infty} \subset \mathcal{X}$ 满足 $\lim_{k \rightarrow \infty} z_k = x$ 以及正标量序列 $\{t_k\}_{k=1}^{\infty}$ ， $t_k \rightarrow 0$ 满足

$$\lim_{k \rightarrow \infty} \frac{z_k - x}{t_k} = d,$$

则向量 d 为 \mathcal{X} 在 x 处的一个切向量所有 x 处切向量的集合称为**切锥**，用 $T_{\mathcal{X}}(x)$ 表示。

Note 4.1.1. 实际上，这里的切锥即为微分几何中的切空间 (tangent space)。

与无约束优化类似，我们要求切锥（可行方向集合）不包含使得目标函数值下降的方向，这就是局部最优点需要满足的必要条件，称为**几何最优性条件**。

Theorem 4.1.1 (几何最优性条件). 假设可行点 $x^* \in \mathcal{X}$ 是问题 4.1 的一个局部极小点, 如果 $f(x)$ 和 $c_i(x)$, $i \in \mathcal{I} \cup \mathcal{E}$ 在点 x^* 处是可微的, 那么有

$$d^T \nabla f(x^*) \geq 0, \quad d \in T_{\mathcal{X}}(x^*)$$

等价于

$$T_{\mathcal{X}}(x^*) \cap \{d \mid \nabla f(x^*)^T d < 0\} = \emptyset$$

Note 4.1.2. 上述定理实际上说的是在最优点 x^* 处, 线性化可行方向和下降方向的交集为空集, 这是显然的。

Definition 4.1.2 (积极集 (active set)). 对于可行点 $x \in \mathcal{X}$, 在该点处的**积极集**定义为两部分指标的集合, 一部分是等式约束对应的指标, 另一部分是不等式约束中在该点等式成立的约束所对应的指标, 即

$$\mathcal{A}(x) = \mathcal{E} \cup \{i \in \mathcal{I} : c_i = 0\}$$

Definition 4.1.3 (线性无关约束品性 LICQ). 给定可行点 $x \in \mathcal{X}$ 及相应的积极集 $\mathcal{A}(x)$. 如果积极集对应的约束函数的梯度, 即 $\nabla c_i(x), i \in \mathcal{A}(x)$, 是线性无关的, 则称**线性无关约束品性 (LICQ)** 在点 x 处成立。

Definition 4.1.4 (线性化可行方向锥). 在点 x 处的**线性化可行方向锥**定义为

$$\mathcal{F}(x) = \left\{ d \mid \begin{array}{l} d^T \nabla c_i(x) = 0, \forall i \in \mathcal{E}, \\ d^T \nabla c_i(x) \geq 0, \forall i \in \mathcal{I} \cap \mathcal{A}(x) \end{array} \right.$$

Lemma 4.1.2. 给定任意可行点 $x \in \mathcal{X}$, 如果在该点处 LICQ 成立, 则有 $T_{\mathcal{X}} = \mathcal{F}(x)$.

直接验证几何最优性条件中交集是空集仍是较为麻烦的, 为此我们介绍一个更加方便的方式

Lemma 4.1.3 (Farkas 引理). 设 p 和 q 是两个非负整数, 给定向量组 $\{a_i \in \mathbb{R}^n, i = 1, 2, \dots, p\}$, $\{b_i \in \mathbb{R}^n, i = 1, 2, \dots, q\}$ 和 $c \in \mathbb{R}^n$. 满足下列条件:

$$d^T a_i = 0, \quad i = 1, 2, \dots, p,$$

$$d^T b_i \geq 0, \quad i = 1, 2, \dots, q,$$

$$d^T c < 0$$

的 d 不存在当且仅当存在一组 $\lambda_i, i = 1, 2, \dots, p$ 和 $\mu_i \geq 0, i = 1, 2, \dots, q$, 使得

$$c = \sum_{i=1}^p \lambda_i a_i + \sum_{i=1}^q \mu_i b_i.$$

利用 Farkas 引理, 我们可以把几何最优性条件 (交集为空集) 写为下面的等价形式:

$$-\nabla f(x^*) = \sum_{i \in \mathcal{E}} \lambda_i^* \nabla c_i(x^*) + \sum_{i \in \mathcal{A}(x^*) \cap \mathcal{I}} \lambda_i^* \nabla c_i(x^*),$$

其中 $\lambda_i^* \in \mathbb{R}, i \in \mathcal{E}, \lambda_i^* \geq 0, i \in \mathcal{A}(x^*) \cap \mathcal{I}$. 如补充定义 $\lambda_i^* = 0, i \in \mathcal{I} \setminus \mathcal{A}(x^*)$, 那么有

$$-\nabla f(x^*) = \sum_{i \in \mathcal{I} \cup \mathcal{E}} \lambda_i^* \nabla c_i(x^*),$$

这恰好为 Lagrange 函数关于 x 的一阶最优性条件, 另外, 对于任意的 $i \in \mathcal{I}$, 我们注意到

$$\lambda_i^* c_i(x^*) = 0.$$

上式称为**互补松弛条件**, 这个条件说明对于不等式约束, 以下两种情况至少出现一种:

1. 乘子 $\lambda_i \geq 0$
2. 不等式约束失效, 即 $c_i(x^*) > 0$ 严格成立

一般来讲, 如果上述两种情况恰好只有一种满足时, 我们称**严格互补松弛条件**成立, 一般来说具有严格互补松弛条件的最优值点有较好的性质。

综上, 我们有如下的一阶必要条件, 也称作 KKT 条件, 并称满足该条件的点 x^*, λ^* 为 **KKT 对**.

Theorem 4.1.4 (Karush-Kuhn-Tucker 条件). 设 x^* 是问题 4.1 的一个局部最优点, 如果

$$T_{\mathcal{X}}(x^*) = \mathcal{F}(x^*)$$

成立, 即在该点 LICQ 成立, 则存在 Lagrange 乘子 λ_i^* 使得下列条件成立:

1. 稳定性条件 $\nabla_x L(x^*, \lambda^*) = \nabla f(x^*) - \sum_{i \in \mathcal{I} \cup \mathcal{E}} \lambda_i^* \nabla c_i(x^*) = 0$,
2. 原始可行性条件 $c_i(x^*) = 0, \forall i \in \mathcal{E}$,
3. 原始可行性条件 $c_i(x^*) \geq 0, \forall i \in \mathcal{I}$,
4. 对偶可行性条件 $\lambda_i^* \geq 0, \forall i \in \mathcal{I}$,
5. 互补松弛条件 $\lambda_i^* c_i(x^*) = 0, \forall i \in \mathcal{I}$

上面的定理是考试经常会涉及的, 所以我们在此通过几道例题加深印象

Example 4.1.1. 求下面约束优化问题的 KKT 点及相应的乘子

$$\begin{aligned} \min f(x) &= x_1^2 + x_2 \\ \text{s.t. } -x_1^2 - x_2^2 + 9 &\geq 0 \\ -x_1 - x_2 + 1 &\geq 0 \end{aligned}$$

相应的 Lagrange 函数为

$$\mathcal{L}(x, \lambda) = x_1^2 + x_2 - \lambda_1(-x_1^2 - x_2^2 + 9) - \lambda_2(-x_1 - x_2 + 1)$$

KKT 条件为

$$\begin{aligned}\frac{\partial \mathcal{L}}{\partial x_1} &= 2x_1 + 2\lambda_1 x_1 + \lambda_2 = 0 \\ \frac{\partial \mathcal{L}}{\partial x_2} &= 1 + 2\lambda_1 x_2 + \lambda_2 = 0 \\ \lambda_1(-x_1^2 - x_2^2 + 9) &= 0, \lambda_1 \geq 0, -x_1^2 - x_2^2 + 9 \geq 0 \\ \lambda_2(-x_1 - x_2 + 1) &= 0, \lambda_2 \geq 0, -x_1 - x_2 + 1 \geq 0\end{aligned}$$

首先 $\lambda_1 = \lambda_2 = 0$ 无解

$\lambda_1 = 0, \lambda_2 \neq 0$ 得 $\lambda_2 = -1$, 矛盾

$\lambda_1 \neq 0, \lambda_2 = 0$ 得 $x = (0, -3)^T, \lambda = (\frac{1}{6}, 0)^T$

$\lambda_1 \neq 0, \lambda_2 \neq 0$ 得 $x = (\frac{1 \pm \sqrt{17}}{2}, \frac{1 \mp \sqrt{17}}{2})^T$, 有 $\lambda_2 = -\frac{1}{2}$ 矛盾。

综上, 只有一个 KKT 点 $(0, -3)^T$ 及相应的 Lagrange 乘子 $(\frac{1}{6}, 0)^T$ 。

Chapter 5

补充内容一：锥及 Karush–Kuhn–Tucker 条件

5.1 凸锥 (Convex Cone)

Definition 5.1.1 (凸锥). 令 $K \subset \mathcal{X}$ 为一凸集, 如果 K 对正数数乘封闭, 即 $\forall x \in K$ 与 $\forall \lambda > 0$, 我们有 $\lambda x \in K$, 我们称 K 为锥.

下面是关于线性代数中的锥的表述, 摘自 Wikipedia, 注意与凸锥对比 (凸组合要属于内部)

Note 5.1.1. 凸锥为一个在有序域上的线性子空间, 关于正数的线性组合封闭。

如下是一些常见的凸锥的例子

Example 5.1.1. 范数锥 (*norm cone*)

$$\{(x, r) \in \mathbb{R}^{d+1} \mid \|x\| \leq r\}$$

Example 5.1.2. 半正定对称锥 (*symmetrical positive semi-definite cone*)

$$\mathcal{S}_+^n = \{A \in \mathcal{S} \mid A \succeq 0\}$$

Example 5.1.3. 条件半正定对称锥 (*conditional symmetrical positive semi-definite cone*)

$$K_+^n = \{X \in \mathcal{S}^n \mid \omega^T X \omega \geq 0, \omega \in \{e\}^\perp\}$$

其中 $e = (1, 1, \dots, 1)^T \in \mathbb{R}^n$

Example 5.1.4. 非负卦限锥 (*Non-negative quadrant cone*)

$$\mathbb{R}_+^n = \{x \in \mathbb{R}^n \mid x_i \geq 0, i = 1, \dots, n\}$$

Example 5.1.5. 正卦限锥 (*Positive quadrant cone*)

$$\{x \in \mathbb{R}^n \mid x_i > 0, i = 1, \dots, n\}$$

Example 5.1.6. 半平面 (*half space*) . 对于任意非零的 $b \in \mathbb{R}^n$ 与 $\beta \in \mathbb{R}$, 下面的集合被称为闭半空间

$$\{x \in \mathbb{R}^n \mid \langle x, b \rangle \leq \beta\}, \quad \{x \in \mathbb{R}^n \mid \langle x, b \rangle \geq \beta\}$$

下面的集合被称为 k 开半空间

$$\{x \in \mathbb{R}^n \mid \langle x, b \rangle < \beta\}, \quad \{x \in \mathbb{R}^n \mid \langle x, b \rangle > \beta\}$$

5.1.1 法锥 (Normal Cone)

Definition 5.1.2. 向量 x^* 被称为在 $a \in C$ 点垂直于凸集 C , 如果 x^* 不与 C 中任意以 a 为短点的线段形成锐角, 即

$$\langle x - a, x^* \rangle \leq 0, \quad \forall x \in C$$

由上面的垂直于 C 的向量 x^* 组成的集合被称为法锥

Definition 5.1.3.

$$N_C(a) \triangleq \begin{cases} \{d \mid \langle d, z - a \rangle \leq 0, \forall z \in C\}, & a \in C \\ \emptyset, & \text{其他} \end{cases}$$

5.1.2 极锥 (Polar Cone)

Definition 5.1.4. 凸锥 $C \subset \mathbb{R}^n$ 的极锥 C° 定义为

$$C^\circ \triangleq \{y \in \mathbb{R}^n \mid \langle y, x \rangle \leq 0, \forall x \in C\}.$$

Note 5.1.2. 即与 C 中所有元素都不构成锐角 (构成直角或钝角)。

Example 5.1.7.

$$K = \{(x, y) \mid y = 2x, x \geq 0\} \subset \mathbb{R}^2$$

$$K^\circ = \{(x, y) \mid y \leq -\frac{x}{2}\}$$

5.1.3 对偶锥 (Dual Cone)

Definition 5.1.5. 锥 C 的对偶锥 C^* 定义为

$$C^* \triangleq \{y \in \mathbb{R}^n \mid \langle y, x \rangle \geq 0, \forall x \in C\}$$

若 $C^* = C$, 则称 C 自对偶 (*self-dual*) .

Note 5.1.3. 与极锥的定义正好相反，对偶锥中的元素和原来锥中的元素不构成钝角（构成直角或锐角）。

Note 5.1.4. 容易发现极锥与对偶锥的关系

$$C^* = -C^\circ$$

这里的相等是指集合相等。

5.2 Lagrange 函数与 KKT 条件

5.2.1 Lagrange 函数

考虑下面的约束优化问题

$$\begin{aligned} \min_{x \in \mathcal{X}} f(x) \\ \text{s.t. } G(x) \in \mathcal{K}. \end{aligned}$$

其中 $f : \mathcal{X} \rightarrow \mathbb{R}, f \in C^1(\mathcal{X}), G : \mathcal{X} \rightarrow \mathcal{Y}, G \in C^1(\mathcal{X})$. 并且 $\mathcal{K} \subset \mathcal{Y}$ 是一个凸锥. 则该问题的 Lagrange 函数为

$$\mathcal{L}(x, \mu) = f(x) - \langle \mu, G(x) \rangle$$

下面是对 KKT 条件的化简

$$\nabla_x \mathcal{L}(\bar{x}, \bar{\mu}) = 0$$

注意到 \bar{x} 是最优解，于是 $\nabla_x f(\bar{x}) = 0$ ，故有

$$0 = \langle \bar{\mu}, G(\bar{x}) \rangle$$

于是，由法锥的定义，有

$$-\bar{\mu} \in N_{\mathcal{K}}(G(\bar{x}))$$

即

$$G(\bar{x}) \in \mathcal{K}, \langle -\bar{\mu}, d - G(\bar{x}) \rangle \leq 0, \forall d \in \mathcal{K}$$

注意到 \mathcal{K} 为凸锥， $0 \in \mathcal{K}$ 且 $2G(\bar{x}) \in \mathcal{K}$. 取 $d = 2G(\bar{x})$ ，得到 $\langle -\bar{\mu}, G(\bar{x}) \rangle \leq 0$ ， $\langle \bar{\mu}, G(\bar{x}) \rangle \geq 0$ ，再取 $d = 0$ ，得到 $\langle \bar{\mu}, G(\bar{x}) \rangle \leq 0$ ，于是

$$\langle \bar{\mu}, G(\bar{x}) \rangle = 0$$

由对偶锥的定义， $\bar{\mu} \in \mathcal{K}^*$.

Note 5.2.1. 上面的推导过程实际上是一般的锥优化问题的对偶参数（Lagrange 乘子）所满足的条件。

下面定义了对偶问题的具体形式

Definition 5.2.1. Lagrange 对偶函数 $g(\mu) = \inf_{x \in \mathcal{X}} \mathcal{L}(x, \mu)$. 故有

$$\sup_{\mu \in \mathcal{K}^*} \inf_{x \in \mathcal{X}} \mathcal{L}(x, \mu) = \sup_{\mu \in \mathcal{K}^*} g(\mu)$$

将原问题转化为了相应的对偶问题

下面是求解对偶问题的具体例子

Example 5.2.1. 考虑线性规划问题

$$\begin{aligned} \min_{x \in \mathbb{R}^n} & \langle c, x \rangle \\ \text{s.t.} & Ax = b, \\ & x \geq 0, \end{aligned}$$

其中 $A \in \mathbb{R}^{m \times n}$, $b \in \mathbb{R}^m$ 是给定的.

$$G(x) = \begin{bmatrix} Ax - b \\ x \end{bmatrix}, \quad \mathcal{K} = \{0\}^m \times \mathbb{R}_+^n$$

注意到

$$\mathcal{K}^* = \mathbb{R}^m \times \mathbb{R}^n$$

令 Lagrange 乘子

$$\mu = \begin{bmatrix} \lambda \\ v \end{bmatrix} \in \mathbb{R}^{m+n}, \lambda \in \mathbb{R}^m, v \in \mathbb{R}^n.$$

于是有

$$\begin{aligned} \mu \in \mathcal{K}^* & \iff \lambda \in \mathbb{R}^m, v \in \mathbb{R}^n \\ G(x) \in \mathcal{K} & \iff Ax - b = 0, x \geq 0. \end{aligned}$$

该问题的 Lagrange 函数为

$$\begin{aligned} \mathcal{L}(x, \mu) &= \langle c, x \rangle - \langle G(x), \mu \rangle \\ &= \langle c, x \rangle - \left\langle \begin{bmatrix} Ax - b \\ x \end{bmatrix}, \begin{bmatrix} \lambda \\ v \end{bmatrix} \right\rangle \\ &= \langle c, x \rangle - \langle Ax - b, \lambda \rangle - \langle x, v \rangle \\ &= \langle x, c - A^T \lambda - v \rangle + \langle b, \lambda \rangle \end{aligned}$$

根据 Lagrange 对偶函数的定义, 我们有

$$\begin{aligned} g(\mu) &= \inf_{x \in \mathbb{R}^n} \mathcal{L}(x, \mu) \\ &= \inf_{x \in \mathbb{R}^n} \langle x, c - A^T \lambda - v \rangle + \langle b, \lambda \rangle \\ &= \langle b, \lambda \rangle + \begin{cases} 0, & c - A^T \lambda - v = 0 \\ -\infty, & \text{其他} \end{cases} \end{aligned}$$

由于我们要求解的是线性规划问题的可行解，只需要考虑 $c - A^T \lambda - v = 0$ 的情况即可

$$\begin{aligned}\sup_{\mu \in \mathcal{K}^*} \inf_{x \in \mathbb{R}^n} \mathcal{L}(x, \mu) &= \sup_{\mu \in \mathcal{K}^*} g(\mu) \\ &= \langle b, \lambda \rangle, \text{ and } c - A^T \lambda - v = 0.\end{aligned}$$

整理即得 LP 的对偶问题

$$\begin{aligned}\max_{\lambda \in \mathbb{R}^m, v \in \mathbb{R}^n} \quad & \langle b, \lambda \rangle \\ \text{s.t.} \quad & c - A^T \lambda - v = 0 \\ & v \geq 0.\end{aligned}$$

Note 5.2.2. 小结：这玩意就离谱，这老师就离谱.

Chapter 6

补充内容二：信赖域算法

本节内容主要参考前面提到的 Algorithms for Optimization by Mykel J. Kochenderfer, Tim A. Wheeler, 外加一些 Wikipedia 中的内容.

6.1 算法简介

信赖域算法 (Trust-region Method) 是解决非线性优化 (NLP) 问题的最重要的数值优化算法之一。它的工作过程是, 首先在当前最佳点 x^k 附近定义一个邻域, 在该邻域内构建一个模型函数 (model function) m_k 来近似 f (通常是 f 在 x^k 处的二阶 Taylor 多项式)。在该信赖域内选择一个步 (包括步长与方向), 如果该次前进之后函数值获得了显著的下降, 那么认为该模型函数是原函数的一个良好近似, 但是若改进微不足道甚至是负面改进, 则我们该模型函数无法准确近似原函数, 同时减小信赖域。

和线搜索算法相同都是借助 Taylor 展开来对目标函数进行局部近似, 但它们看待近似函数的方式不同。在线搜索算法中, 我们先利用近似模型求出下降方向, 然后给定步长; 而在信赖域类算法中, 我们直接在一个有界区域内求解这个近似模型, 而后迭代到下一个点。因此信赖域算法实际上是同时选择了方向和步长。

6.2 信赖域算法框架

由带 Lagrange 余项的 Taylor 公式, 有

$$f(x^k + d) = f(x^k) + \nabla f(x^k)^T d + \frac{1}{2} d^T \nabla^2 f(x^k + td) d,$$

其中 $t \in (0, 1)$. 我们用 f 的二阶近似来刻画 $f(x)$ 在 $x = x^k$ 的性质:

$$m_k(d) = f(x^k) + \nabla f(x^k)^T d + \frac{1}{2} d^T B^k d, \quad (6.1)$$

B^k 是对称矩阵, 要求 B^k 是 Hessian 矩阵的近似. 若 B^k 恰好是 Hessian 矩阵, 则 $m_k(d)$ 的近似误差为 $\mathcal{O}(\|d\|^3)$.

Figure 6.1
信赖域方法

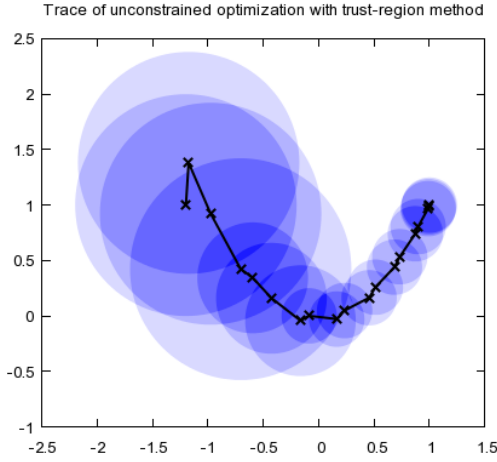
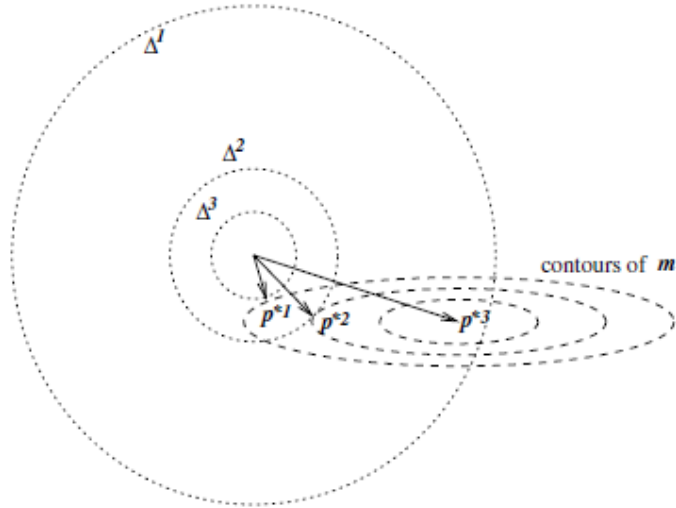


Figure 6.2
步长和改进方向都是预定信赖域大小的结果



考虑到 Taylor 展开是函数的局部性质，于是我们仅在下面的求内考虑问题

$$\Omega_k = \{x^k + d \mid \|d\| \leq \Delta_k\}$$

其中 Δ_k 是一个与迭代有关的参数. 我们称 Ω_k 为**信赖域**, Δ_k 为**信赖域步长**. 我们相信在信赖域中 $m_k(d)$ 能够很好地近似 $f(x)$, 因此信赖域算法每一步都要解决如下的子问题:

$$\min_{d \in \mathbb{R}^n} m_k(d), \quad s.t. \quad \|d\| \leq \Delta_k. \quad (6.2)$$

事实上，在信赖域算法中，选取信赖域半径至关重要，它决定了算法的收敛性. 考虑到信赖域半径是“对模型函数 $m_k(d)$ 相信的程度”，那么如果 $m_k(d)$ 对函数 $f(x)$ 近似较好，我们应扩大信赖域半径，反之就应该减小信赖域重新计算. 我们引入如下的定义来衡量模型函数 $m_k(d)$ 近似

的好坏：

$$\rho_k = \frac{f(x^k) - f(x^k + d)}{m_k(0) - m_k(d^k)}, \quad (6.3)$$

其中 d^k 为求解信赖域问题子问题(6.2)得到的迭代方向. 根据 ρ_k 的定义我们知道它即为函数值实际下降量与预估下降量的比值. 如果 ρ_k 接近 1, 说明 $m_k(d)$ 近似 $f(x)$ 是比较成功的, 我们应该扩大 Δ_k , 否则应该缩小信赖域. 下面我们给出具体的算法, 注意我们其实回避了如何求解子问题(6.2)

Algorithm 7 信赖域算法

Require: 最大半径 Δ_{max} , 初始半径 Δ_0 , 初始点 x^0 , $k \leftarrow 0$.

Require: 给定参数 $0 \leq \eta < \bar{\rho}_1 < \bar{\rho}_2 < 1$, $\gamma_1 < 1 < \gamma_2$.

```

1: while 未达到收敛准则 do
2:   计算信赖域问题子问题(6.2)得到迭代方向  $d^k$  (实际上是步长 + 方向).
3:   根据(6.3)计算下降率  $\rho_k$ .
4:   if  $\rho_k < \bar{\rho}_1$  then
5:     缩小信赖域半径:  $\Delta_{k+1} \leftarrow \gamma_1 \Delta_k$ .
6:   else
7:     if  $\rho_k > \bar{\rho}_2$  以及  $\|d^k\| = \Delta_k$  then
8:       扩大信赖域半径:  $\Delta_{k+1} \leftarrow \min\{\gamma_2 \Delta_k, \Delta_{max}\}$ .
9:     else
10:      信赖域半径不变:  $\Delta_{k+1} \leftarrow \Delta_k$ 
11:    end if
12:  end if
13:  if  $\rho_k > \eta$  then
14:    更新  $x^{k+1} \leftarrow x^k + d^k$ .
15:  else
16:     $x^{k+1} \leftarrow x^k$ .
17:  end if
18:   $k \leftarrow k + 1$ .
19: end while
    
```

实际上, 信赖域算法的大致思想就是首先判断模型函数 $m_k(d)$ 是否是原函数 $f(x)$ 在 Δ_k 中的良好近似, 并根据此修改信赖域半径. 接着迈出步子走就好了.

我们通常取参数如下: $\bar{\rho}_1 = 0.25$, $\bar{\rho}_2 = 0.75$ 以及 $\gamma_1 = 0.25$, $\gamma_2 = 2$.

下面我们给出求解信赖域子问题(6.2)的算法. 而这才是信赖域算法的关键.

6.3 信赖域算法子问题求解

信赖域子问题(6.2)仅仅是一个涉及二次函数的约束优化问题，我们可以通过 KKT 条件求解，通过 KKT 条件得到最优解 d^* 应满足如下条件（过程太长，详情参见文再文老师的书）：

Theorem 6.3.1. d^* 是信赖域子问题

$$\min \quad m(d) = f + g^T d + \frac{1}{2} d^T B d, \quad s.t. \quad \|d\| \leq \Delta$$

的全局极小解当且仅当 d^* 可行且存在 $\lambda \geq 0$ 满足

$$\begin{aligned} (B + \lambda I)d^* &= -g, \\ \lambda(\Delta - \|d^*\|) &= 0, \\ (B + \lambda I) &\succeq 0. \end{aligned} \tag{6.4}$$

寻找子问题(6.2)的近似解常有三种方法

1. Cauchy 点
2. Dogleg 算法（针对 $B_k \succ 0$ ）
3. 二维子空间极小化（对于 B_k 非正定）

首先我们介绍 Cauchy 点

6.3.1 Cauchy 点

Definition 6.3.1 (Cauchy 点). 设 $m_k(d)$ 是 $f(x)$ 在点 $x = x^k$ 处的二阶近似，常数 τ_k 为如下优化问题的解

$$\begin{aligned} \min \quad & m_k(-\tau \nabla f(x^k)), \\ s.t. \quad & \|\tau \nabla f(x^k)\| \leq \Delta_k, \tau \geq 0 \end{aligned}$$

称 $x_C^k := x^k + d_C^k$ 为 Cauchy 点，其中 $d_C^k = -\tau_k \nabla f(x^k)$.

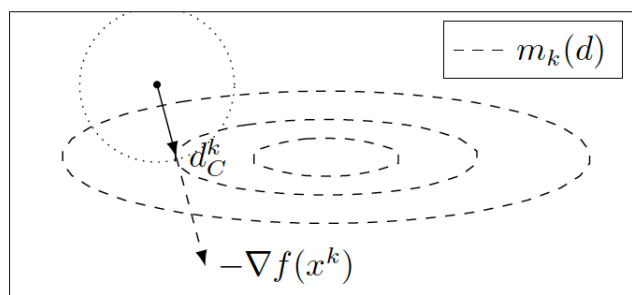
实际上，根据 Cauchy 点的定义，它实际上是对 $m_k(d)$ 进行一次精确线搜索的梯度法，只不过这个线搜索是考虑了信赖域约束的，参见下图 事实上，根据定义，Cauchy 点能够显式地计算出来，为方便起见，我们用 g^k 表示 $\nabla f(x^k)$

$$\tau_k = \begin{cases} \frac{\Delta_k}{\|g^k\|}, & (g^k)^T B^k g^k \leq 0, \\ \min\left\{\frac{\|g^k\|^2}{(g^k)^T B^k g^k}, \frac{\Delta_k}{\|g^k\|}\right\}, & \text{其它.} \end{cases}$$

6.3.2 dog leg 法

Dog let 法由 Powell 提出，结合了 Gauss–Newton 算法和梯度下降法，但是使用了显式信任区域。在每次迭代中，如果高斯-牛顿算法中的步骤位于信任区内，则将其用于更新当前解决方案。

Figure 6.3
Cauchy 点



如果不是，则算法沿最陡的下降方向（称为柯西点）搜索目标函数的最小值。如果柯西点在信任区域之外，则将其截断到信任区域的边界，并将其作为新的解决方案。如果柯西点在信任区域内，则在信任区域边界与连接柯西点和 Gauss-Newton 步（dog leg step）的线之间的交点处采用新解。

Note 6.3.1. 我真的找不到太多 *dog leg* 法的资料，它的 *slide* 味同嚼蜡看不懂。以后有心情再了解。